



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE

United States Patent and Trademark Office

Address: COMMISSIONER FOR PATENTS

P.O. Box 1450

Alexandria, Virginia 22313-1450

www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/710,356	07/02/2004	Steven T. Shaughnessy	BORI/0222.00	4355
28653	7590	09/02/2008		
JOHN A. SMART				
201 LOS GATOS				
SARATOGA RD, #161				
LOS GATOS, CA 95030-5308				
EXAMINER				
SANDERS, AARON J				
ART UNIT		PAPER NUMBER		
2168				
MAIL DATE		DELIVERY MODE		
09/02/2008		PAPER		

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary

Application No.

10/710,356

Applicant(s)

SHAUGHNESSY, STEVEN T.

Examiner

AARON SANDERS

Art Unit

2168

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --
Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

- 1) ☒ Responsive to communication(s) filed on 06 June 2008.
- 2a) ☒ This action is **FINAL**. 2b) ☐ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

- 4) ☒ Claim(s) 1-3, 5-8, 10-19, 21-24 and 26-30 is/are pending in the application.
- 4a) Of the above claim(s) _____ is/are withdrawn from consideration.
- 5) ☐ Claim(s) _____ is/are allowed.
- 6) ☒ Claim(s) 1-3, 5-8, 10-19, 21-24 and 26-30 is/are rejected.
- 7) ☐ Claim(s) _____ is/are objected to.
- 8) ☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☒ The drawing(s) filed on 02 July 2004 is/are: a) ☐ accepted or b) ☒ objected to by the Examiner.
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some * c) ☐ None of:
1. ☐ Certified copies of the priority documents have been received.
 2. ☐ Certified copies of the priority documents have been received in Application No. _____.
 3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

- 1) ☒ Notice of References Cited (PTO-892)
- 2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
- 3) ☐ Information Disclosure Statement(s) (PTO/SB/08)
Paper No(s)/Mail Date _____
- 4) ☐ Interview Summary (PTO-413)
Paper No(s)/Mail Date _____
- 5) ☐ Notice of Informal Patent Application
- 6) ☐ Other: _____

DETAILED ACTION

Response to Amendment

The amendment filed 6 June 2008 has been entered. Claims 1-3, 5-8, 10-19, 21-24, and 26-30 are pending. Claims 1-2, 5, 8, 10-12, 17-18, 21, 24, and 26-28 are currently amended. Claims 4, 9, 20, and 25 are cancelled. No claims are new. This action is FINAL, as necessitated by amendment.

Drawings

The drawings are objected to under 37 CFR 1.83(a). The drawings must show every feature of the invention specified in the claims. Therefore, the method of claim 1 must be shown or the features canceled from the claims. No new matter should be entered.

Corrected drawing sheets in compliance with 37 CFR 1.121(d) are required in reply to the Office action to avoid abandonment of the application. Any amended replacement drawing sheet should include all of the figures appearing on the immediate prior version of the sheet, even if only one figure is being amended. The figure or figure number of an amended drawing should not be labeled as "amended." If a drawing figure is to be canceled, the appropriate figure must be removed from the replacement sheet, and where necessary, the remaining figures must be renumbered and appropriate changes made to the brief description of the several views of the drawings for consistency. Additional replacement sheets may be necessary to show the renumbering of the remaining figures. Each drawing sheet submitted after the filing date of an application must be labeled in the top margin as either "Replacement Sheet" or "New Sheet" pursuant to 37 CFR 1.121(d). If the changes are not accepted by the examiner, the applicant will

be notified and informed of any required corrective action in the next Office action. The objection to the drawings will not be held in abeyance.

Claim Objections

Claims 3 and 19 are objected to under 37 CFR 1.75(c), as being of improper dependent form for failing to further limit the subject matter of a previous claim. Applicant is required to cancel the claim(s), or amend the claim(s) to place the claim(s) in proper dependent form, or rewrite the claim(s) in independent form.

As per claim 17, the phrase “creating write view” should be “creating a write view.”

Claim Rejections - 35 USC § 112

The following is a quotation of the first paragraph of 35 U.S.C. 112:

The specification shall contain a written description of the invention, and of the manner and process of making and using it, in such full, clear, concise, and exact terms as to enable any person skilled in the art to which it pertains, or with which it is most nearly connected, to make and use the same and shall set forth the best mode contemplated by the inventor of carrying out his invention.

Claims 1 and 17 are rejected under 35 U.S.C. 112, first paragraph, as failing to comply with the written description requirement. The claim(s) contains subject matter which was not described in the specification in such a way as to reasonably convey to one skilled in the relevant art that the inventor(s), at the time the application was filed, had possession of the claimed invention.

Specifically, the “creating a write view” does not appear in the specification. The “write cache view” of par. 47 may be the claimed “write view,” but this is not clear.

Claim Rejections - 35 USC § 103

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

Claims 1-3, 5-8, 10, 12-13, 15-19, 21-24, 26, and 28-29 are rejected under 35 U.S.C. 103(a) as being unpatentable over Klein et al., U.S. 6,631,374, in view of Hayashi et al., U.S. 5,715,447, and in view of DeWitt, Jr. et al., U.S. 7,093,081.

1. Klein teaches “*In a database system employing a transaction log, an improved method for restoring databases to a consistent version supporting read-only uses, the method comprising,*” see col. 7, line 66 – col. 8, line 8, “FIG. 9 is a flow diagram showing a method for providing fine-grained temporal database access 140 in a consistent read environment.”

Klein teaches “*in response to a read-only transaction of the given database, creating a read-only cache view of the given database using the given database’s transaction log by logically undoing transactions which have begun but have yet to commit, said read-only cache view comprising particular database blocks in the shared cache that record a transactionally consistent version of the given database at a given point in time supporting read-only uses,*” see Fig. 10, col. 2, lines 16-22, “Some database systems incorporate transaction logs which track and record all operations performed against the database. Log mining allows those operations which have effected the data to be reconstructed back into database statements,” col. 8, lines 30-53, “FIG. 10 is a flow diagram showing a routine for performing a consistent read operation 160... First, the relative database block is retrieved (block 161) from the persistent storage 22. Next,

each interested transaction entry (ITE) 51 (shown in FIG. 3) is iteratively processed (blocks 162-169) to logically reconstruct the database 23 as of the requested query time... For each ITE 51, transactions are rolled back in an ordered manner... Thus, if the transaction referenced by the transaction identifier xid is active (block 163), the entire transaction is rolled back to logically undo the transaction (block 164),” where the claimed “cache view” would be the referenced reconstructed database.

Klein teaches “*and performing the read-only transaction using the read-only cache view and returning results of the read-only transaction, without blocking performance of transactions involving write operations using the write view of the given database,*” see Fig. 10, col. 9, lines 5-17, “The retrieved data values from the rolled back transactions are provided (block 169),” and col. 7, line 66 – col. 8, line 8, “As would be readily recognized by one skilled in the art, a typical database engine could be concurrently processing multiple transaction operations at any given time. The described invention implicates read operations and does not directly effect the processing of other forms of database transactions.”

Klein does not teach “*providing a shared cache storing database blocks in memory of the database system.*” Hayashi does, however, see Fig. 4 and col. 1, lines 30-41, “When a transaction accesses the database for some data through the database management system, the data may already be in a buffer shared by transactions due to another transaction that previously accessed the same data, or the data must be transferred from the database to the shared buffer.” Thus, it would have been obvious to one of ordinary skill in the database art at the time of the invention to combine the teachings of the cited references because Hayashi’s teachings would

have allowed Klein's method to gain a common means for storing frequently used database blocks, see col. 1, lines 30-41.

Klein does not teach *"creating a write view of a given database in the shared cache supporting read and write uses of the given database."* Hayashi does, however, see Fig. 4 and col. 1, lines 30-41, "When a transaction accesses the database for some data through the database management system, the data may already be in a buffer shared by transactions due to another transaction that previously accessed the same data, or the data must be transferred from the database to the shared buffer," where the claimed "write view" is the referenced view of the "shared buffer." Thus, it would have been obvious to one of ordinary skill in the database art at the time of the invention to combine the teachings of the cited references because Hayashi's teachings would have allowed Klein's method to gain a common means for storing frequently used database blocks, see col. 1, lines 30-41.

Klein does not teach *"temporarily storing any database blocks that overflow said shared cache in a temporary database during use of the read-only cache view by the read-only transaction."* DeWitt does, however, see col. 4, lines 3-21, "However, if there is not sufficient space in the cache to store the block of instructions/data that is to be reloaded, then the block of instructions/data, or at least the overflow portion of the block of instructions/data, is loaded into a reserved portion of cache," where the claimed "temporary database" is the referenced "reserved portion of cache." Thus, it would have been obvious to one of ordinary skill in the database art at the time of the invention to combine the teachings of the cited references because DeWitt's teachings would have allowed Klein's method to store blocks of data that overflow the cache, see col. 4, lines 3-21.

2. Klein does not teach “*The method of claim 1, wherein during occurrence of the read-only transaction any database blocks associated with the read-only cache view are not written from the shared cache to the given database.*” Hayashi does, however, see Fig. 1 and col. 3, line 66 to col. 4, line 7, “The contents (i.e., pages) of the shared buffer are written back to the disk (i.e., data base 20) at a predetermined timing. Then, a log holding data updated by the last read or write operation will not be needed for recovering the contents of the buffer even if they are lost. The log buffer 16 temporarily stores pre-update and post-update logs of the shared buffers B1 and B2. The contents of the log buffer 16 are nonvolatilized by saving them in the log file 19 at a predetermined timing.” Thus, it would have been obvious to one of ordinary skill in the database art at the time of the invention to combine the teachings of the cited references because Hayashi’s teachings would have allowed Klein’s method to gain a common means for storing frequently used database blocks, see col. 1, lines 30-41.

3. Klein does not teach “*The method of claim 1, wherein the blocks that overflow the shared cache are temporarily stored in a temporary database.*” DeWitt does, however, see col. 4, lines 3-21, “However, if there is not sufficient space in the cache to store the block of instructions/data that is to be reloaded, then the block of instructions/data, or at least the overflow portion of the block of instructions/data, is loaded into a reserved portion of cache,” where the claimed “temporary database” is the referenced “reserved portion of cache.” Thus, it would have been obvious to one of ordinary skill in the database art at the time of the invention to combine the teachings of the cited references because DeWitt’s teachings would have allowed Klein’s method to store blocks of data that overflow the cache, see col. 4, lines 3-21.

5. Klein does not teach *“The method of claim 1, wherein the temporary database is used only in the event the read-only cache view overflows the shared cache.”* DeWitt does, however, see col. 4, lines 3-21, “However, if there is not sufficient space in the cache to store the block of instructions/data that is to be reloaded, then the block of instructions/data, or at least the overflow portion of the block of instructions/data, is loaded into a reserved portion of cache,” where the claimed “temporary database” is the referenced “reserved portion of cache.” Thus, it would have been obvious to one of ordinary skill in the database art at the time of the invention to combine the teachings of the cited references because DeWitt’s teachings would have allowed Klein’s method to store blocks of data that overflow the cache, see col. 4, lines 3-21.

6. Klein does not teach *“The method of claim 1, further comprising: using an allocation bitmap for indicating database blocks temporarily stored in the temporary database.”* Hayashi does, however, see Fig. 2, “bit map 30” and “overflow pages 31.” Thus, it would have been obvious to one of ordinary skill in the database art at the time of the invention to combine the teachings of the cited references because Hayashi’s teachings would have allowed Klein’s method to gain control of the overflow pages, see col. 4, lines 8-18.

7. Klein does not teach *“The method of claim 6, further comprising: upon completion of the read-only transaction, deleting the blocks temporarily stored in the temporary database by updating the allocation bitmap.”* Hayashi does, however, see col. 4, lines 8-18, “any bit of the bit map 30 will be ON when a corresponding one of the overflow pages 31 is in use and OFF when the corresponding page is unused.” Thus, it would have been obvious to one of ordinary skill in the database art at the time of the invention to combine the teachings of the cited

references because Hayashi's teachings would have allowed Klein's method to gain control of the overflow pages, see col. 4, lines 8-18.

8. Klein does not teach "*The method of claim 1, wherein said step of temporarily storing database blocks overflowing the shared cache includes storing a mapping to said database blocks in a table in the temporary database including a first column for maintaining a block number of a read-only cache view block having undo/redo records applied to it and a second column for maintaining a block number allocated to temporarily store said database blocks overflowing the shared cache.*" Hayashi does, however, see col. 5, lines 30-46, "The database contains page data and a table showing relationships between page numbers and locations on the disk." Thus, it would have been obvious to one of ordinary skill in the database art at the time of the invention to combine the teachings of the cited references because Hayashi's teachings would have allowed Klein's method to gain control of the overflow pages, see col. 4, lines 8-18.

10. Klein teaches "*The method of claim 1, further comprising: upon termination of the read-only transaction, marking the read-only cache view as closed,*" see col. 9, lines 5-17, "The retrieved data values from the rolled back transactions are provided (block 169)."

12. Klein teaches "*The method of claim 1, further comprising: sharing the read-only cache view created for the read-only transaction with other read-only transactions which start within a specified period of time following the start of the read-only transaction,*" see col. 9, lines 5-17, "The retrieved data values from the rolled back transactions are provided (block 169)."

13. Klein teaches “*The method of claim 1, further comprising: detecting the read-only transaction,*” see col. 8, lines 30-53, “FIG. 10 is a flow diagram showing a routine for performing a consistent read operation 160.”

Klein teaches “*and upon occurrence of write operations, adding back link log records to the database’s transaction log that serve to link together log records of the transaction log that pertain to a write transaction,*” see col. 2, lines 16-22, “Some database systems incorporate transaction logs which track and record all operations performed against the database. Log mining allows those operations which have effected the data to be reconstructed back into database statements.”

15. Klein teaches “*A computer-readable medium having processor-executable instructions for performing the method of claim 1,*” see col. 6, lines 24-34, “The various implementations of the source code and object and byte codes can be held on a computer-readable storage medium.”

16. Klein teaches “*The method of claim 1, further comprising: downloading a set of processor-executable instructions for performing the method of claim 1,*” see col. 6, lines 24-34, “The various implementations of the source code and object and byte codes can be... embodied on a transmission medium in a carrier wave.”

17. Klein teaches “*A database system capable of restoring databases to a consistent version supporting read-only uses, the system comprising,*” see col. 7, line 66 – col. 8, line 8, “FIG. 9 is a flow diagram showing a method for providing fine-grained temporal database access 140 in a consistent read environment.”

Klein teaches “a computer having a processor and memory,” see Fig. 1 and col. 4, lines 37-43, “The system 10 consists of a database server 11 operating on a host computer system that provides database access to a plurality of clients.”

Klein teaches “a log manager module which manages a transaction log of the database system,” see Fig. 1 and col. 4, lines 37-43, “The system 10 consists of a database server 11 operating on a host computer system that provides database access to a plurality of clients.”

Klein teaches “and a read-only cache view of a given database using the transaction log of the given database, said read-only cache view being created in response to a read-only transaction of the given database, said read-only cache view comprising particular database blocks of the shared cache that record a view of a particular version of the given database at a given point in time... and invokes a transaction manager module for logically undoing transactions which have begun but have yet to commit in creating the read-only cache view... and a transaction manager module for logically undoing transactions which have begun but have yet to commit upon starting the read-only transaction in order to construct the read-only cache view comprising a transactionally consistent prior version of the given database,” see Fig. 10, col. 2, lines 16-22, “Some database systems incorporate transaction logs which track and record all operations performed against the database. Log mining allows those operations which have effected the data to be reconstructed back into database statements,” col. 8, lines 30-53, “FIG. 10 is a flow diagram showing a routine for performing a consistent read operation 160... First, the relative database block is retrieved (block 161) from the persistent storage 22. Next, each interested transaction entry (ITE) 51 (shown in FIG. 3) is iteratively processed (blocks 162-169) to logically reconstruct the database 23 as of the requested query time... For each ITE 51,

transactions are rolled back in an ordered manner... Thus, if the transaction referenced by the transaction identifier xid is active (block 163), the entire transaction is rolled back to logically undo the transaction (block 164),” where the claimed “cache view” is the referenced reconstructed database.

Klein teaches “*performing the read-only transaction using the read-only cache view without blocking performance of transactions involving write operations using the write view of the given database, and returning results of the read-only transaction*,” see Fig. 10, col. 9, lines 5-17, “The retrieved data values from the rolled back transactions are provided (block 169),” and col. 7, line 66 – col. 8, line 8, “As would be readily recognized by one skilled in the art, a typical database engine could be concurrently processing multiple transaction operations at any given time. The described invention implicates read operations and does not directly effect the processing of other forms of database transactions.”

Klein does not teach “*a cache manager module for managing a shared cache that stores database blocks in memory of the database system*.” Hayashi does, however, see Fig. 4 and col. 1, lines 30-41, “When a transaction accesses the database for some data through the database management system, the data may already be in a buffer shared by transactions due to another transaction that previously accessed the same data, or the data must be transferred from the database to the shared buffer.” Thus, it would have been obvious to one of ordinary skill in the database art at the time of the invention to combine the teachings of the cited references because Hayashi’s teachings would have allowed Klein’s method to gain a common means for storing frequently used database blocks, see col. 1, lines 30-41.

Klein does not teach “*and creating write view of a given database in the shared cache supporting read and write uses of the given database.*” Hayashi does, however, see Fig. 4 and col. 1, lines 30-41, “When a transaction accesses the database for some data through the database management system, the data may already be in a buffer shared by transactions due to another transaction that previously accessed the same data, or the data must be transferred from the database to the shared buffer,” where the claimed “write view” is the referenced view of the “shared buffer.” Thus, it would have been obvious to one of ordinary skill in the database art at the time of the invention to combine the teachings of the cited references because Hayashi’s teachings would have allowed Klein’s method to gain a common means for storing frequently used database blocks, see col. 1, lines 30-41.

Klein does not teach “*wherein the cache manager utilizes a temporary database for storing any database blocks that overflow said shared cache during use of the read-only cache view by the read-only transaction.*” DeWitt does, however, see col. 4, lines 3-21, “However, if there is not sufficient space in the cache to store the block of instructions/data that is to be reloaded, then the block of instructions/data, or at least the overflow portion of the block of instructions/data, is loaded into a reserved portion of cache,” where the claimed “temporary database” is the referenced “reserved portion of cache.” Thus, it would have been obvious to one of ordinary skill in the database art at the time of the invention to combine the teachings of the cited references because DeWitt’s teachings would have allowed Klein’s method to store blocks of data that overflow the cache, see col. 4, lines 3-21.

18. Klein does not teach “*The system of claim 17, wherein during occurrence of the read-only transaction any database blocks associated with the read-only cache view are not written*

from the shared cache to the given database.” Hayashi does, however, see Fig. 1 and col. 3, line 66 to col. 4, line 7, “The contents (i.e., pages) of the shared buffer are written back to the disk (i.e., data base 20) at a predetermined timing. Then, a log holding data updated by the last read or write operation will not be needed for recovering the contents of the buffer even if they are lost. The log buffer 16 temporarily stores pre-update and post-update logs of the shared buffers B1 and B2. The contents of the log buffer 16 are nonvolatilized by saving them in the log file 19 at a predetermined timing.” Thus, it would have been obvious to one of ordinary skill in the database art at the time of the invention to combine the teachings of the cited references because Hayashi’s teachings would have allowed Klein’s method to gain a common means for storing frequently used database blocks, see col. 1, lines 30-41.

19. Klein does not teach *“The system of claim 17, wherein the cache manager stores database blocks that overflow the shared cache in a temporary database.”* DeWitt does, however, see col. 4, lines 3-21, “However, if there is not sufficient space in the cache to store the block of instructions/data that is to be reloaded, then the block of instructions/data, or at least the overflow portion of the block of instructions/data, is loaded into a reserved portion of cache,” where the claimed “temporary database” is the referenced “reserved portion of cache.” Thus, it would have been obvious to one of ordinary skill in the database art at the time of the invention to combine the teachings of the cited references because DeWitt’s teachings would have allowed Klein’s method to store blocks of data that overflow the cache, see col. 4, lines 3-21.

21. Klein does not teach *“The system of claim 17, wherein the temporary database is used only in the event the read-only cache view overflows the shared cache.”* DeWitt does, however, see col. 4, lines 3-21, “However, if there is not sufficient space in the cache to store the

block of instructions/data that is to be reloaded, then the block of instructions/data, or at least the overflow portion of the block of instructions/data, is loaded into a reserved portion of cache,” where the claimed “temporary database” is the referenced “reserved portion of cache.” Thus, it would have been obvious to one of ordinary skill in the database art at the time of the invention to combine the teachings of the cited references because DeWitt’s teachings would have allowed Klein’s method to store blocks of data that overflow the cache, see col. 4, lines 3-21.

22. Klein does not teach *“The system of claim 17, wherein said cache manager maintains an allocation bitmap indicating database blocks temporarily stored in the temporary database.”* Hayashi does, however, see Fig. 2, “bit map 30” and “overflow pages 31.” Thus, it would have been obvious to one of ordinary skill in the database art at the time of the invention to combine the teachings of the cited references because Hayashi’s teachings would have allowed Klein’s method to gain control of the overflow pages, see col. 4, lines 8-18.

23. Klein does not teach *“The system of claim 22, wherein said cache manager deletes blocks from the temporary database by updating the allocation bitmap.”* Hayashi does, however, see col. 4, lines 8-18, “any bit of the bit map 30 will be ON when a corresponding one of the overflow pages 31 is in use and OFF when the corresponding page is unused.” Thus, it would have been obvious to one of ordinary skill in the database art at the time of the invention to combine the teachings of the cited references because Hayashi’s teachings would have allowed Klein’s method to gain control of the overflow pages, see col. 4, lines 8-18.

24. Klein does not teach *“The system of claim 17, wherein said cache manager stores a mapping to database blocks overflowing the shared cache in a table of the temporary database including a first column for maintaining a block number of a read- only cache view block having*

undo/redo records applied to it and a second column for maintaining a block number allocated to temporarily store said database blocks overflowing the shared cache.” Hayashi does, however, see col. 5, lines 30-46, “The database contains page data and a table showing relationships between page numbers and locations on the disk.” Thus, it would have been obvious to one of ordinary skill in the database art at the time of the invention to combine the teachings of the cited references because Hayashi’s teachings would have allowed Klein’s method to gain control of the overflow pages, see col. 4, lines 8-18.

26. Klein teaches “*The system of claim 17, wherein said cache manager marks the read-only cache view as closed, upon termination of the read-only transaction,*” see col. 9, lines 5-17, “The retrieved data values from the rolled back transactions are provided (block 169).”

28. Klein teaches “*The system of claim 17, wherein said cache manager shares the read-only cache view created for the read-only transaction with other read-only transactions which start within a specified period of time following the start of the read-only transaction,*” see col. 9, lines 5-17, “The retrieved data values from the rolled back transactions are provided (block 169).”

29. Klein teaches “*The system of claim 17, wherein said log manager detects the read-only transaction, and adds back link log records to the transaction log that serve to link together log records of the transaction log that pertain to a write transaction that may need to be logically undone,*” see col. 8, lines 30-53, “FIG. 10 is a flow diagram showing a routine for performing a consistent read operation 160” and col. 2, lines 16-22, “Some database systems incorporate transaction logs which track and record all operations performed against the

database. Log mining allows those operations which have effected the data to be reconstructed back into database statements.”

Claims 11 and 27 are rejected under 35 U.S.C. 103(a) as being unpatentable over Klein et al., U.S. 6,631,374, in view of Hayashi et al., U.S. 5,715,447, in view of DeWitt, Jr. et al., U.S. 7,093,082, and further in view of *The Authoritative Dictionary of IEEE Standards Terms, Seventh Edition*, IEEE Press, 2000.

11. Klein does not teach “*The method of claim 10, further comprising: when new block allocations need to be made in the shared cache, traversing the shared cache looking for database blocks to purge; and purging database blocks from any read-only cache view that have been marked as closed.*” IEEE does, however, see “garbage collection (B) A database reorganization technique in which the contents of a database are made more compact by physically deleting garbage such as records that have been deleted logically but remain physically in the database” and “cache (2) A small portion of high-speed memory used for temporary storage of frequently-used data, instructions, or operands.” Thus, it would have been obvious to one of ordinary skill in the database art at the time of the invention to combine the teachings of the cited references because IEEE’s teachings would have allowed Klein’s method and system to gain the ability to compact the contents of the database view, see “garbage collection (B) A database reorganization technique in which the contents of a database are made more compact by physically deleting garbage such as records that have been deleted logically but remain physically in the database.”

27. Klein does not teach *“The system of claim 26, wherein said cache manager traverses the shared cache looking for database blocks to purge, and purges database blocks from any read-only view that have been marked as closed when new block allocations need to be made in the shared cache.”* IEEE does, however, see “garbage collection (B) A database reorganization technique in which the contents of a database are made more compact by physically deleting garbage such as records that have been deleted logically but remain physically in the database” and “cache (2) A small portion of high-speed memory used for temporary storage of frequently-used data, instructions, or operands.” Thus, it would have been obvious to one of ordinary skill in the database art at the time of the invention to combine the teachings of the cited references because IEEE’s teachings would have allowed Klein’s method and system to gain the ability to compact the contents of the database view, see “garbage collection (B) A database reorganization technique in which the contents of a database are made more compact by physically deleting garbage such as records that have been deleted logically but remain physically in the database.”

Claims 14 and 30 are rejected under 35 U.S.C. 103(a) as being unpatentable over Klein et al., U.S. 6,631,374, in view of Hayashi et al., U.S. 5,715,447, in view of DeWitt, Jr. et al., U.S. 7,093,082, and further in view of Raz, U.S. 5,701,480.

14. Klein does not teach *“The method of claim 13, further comprising: if the read-only transaction must be undone, using the back link log records to skip portions of the transaction log that are irrelevant for undoing an uncommitted write transaction, wherein the back link log records are only generated in the transaction log when there are active read only transactions.”* Raz does, however, see col. 62, lines 3-17, “the computer 20 processes transactions using an

‘undo’ recovery mechanism that provides very fast recovery because only the effects of failed transactions must be undone.” Thus, it would have been obvious to one of ordinary skill in the database art at the time of the invention to combine the teachings of the cited references because Raz’s teachings would have allowed Klein’s method and system to gain greater efficiency by not undoing redundant or unnecessary transactions, see Raz col. 62, lines 3-17.

30. Klein does not teach *“The system of claim 29, wherein said log manager uses the back link log records to skip portions of the transaction log that are irrelevant for undoing the write transaction, wherein the back link log records are only generated in the transaction log when there are active read only transactions.”* Raz does, however, see col. 62, lines 3-17, “the computer 20 processes transactions using an ‘undo’ recovery mechanism that provides very fast recovery because only the effects of failed transactions must be undone.” Thus, it would have been obvious to one of ordinary skill in the database art at the time of the invention to combine the teachings of the cited references because Raz’s teachings would have allowed Klein’s method and system to gain greater efficiency by not undoing redundant or unnecessary transactions, see Raz col. 62, lines 3-17.

Response to Arguments

As per Applicant’s argument that the drawings show all the features of claim 1, the Examiner respectfully disagrees. Specifically, at least the steps of “providing a shared cache,” “creating a write view,” “creating a read-only cache view,” “temporarily storing,” and “performing the read-only transaction” do not appear in the figures.

As per Applicant's argument that Hayashi does not teach "creating a read-only cache view of the given database using the given database's transaction log," the Examiner agrees. The Examiner has instead cited Klein Fig. 10, col. 2, lines 16-22, "Some database systems incorporate transaction logs which track and record all operations performed against the database. Log mining allows those operations which have effected the data to be reconstructed back into database statements," col. 8, lines 30-53, "FIG. 10 is a flow diagram showing a routine for performing a consistent read operation 160... First, the relative database block is retrieved (block 161) from the persistent storage 22. Next, each interested transaction entry (ITE) 51 (shown in FIG. 3) is iteratively processed (blocks 162-169) to logically reconstruct the database 23 as of the requested query time... For each ITE 51, transactions are rolled back in an ordered manner... Thus, if the transaction referenced by the transaction identifier xid is active (block 163), the entire transaction is rolled back to logically undo the transaction (block 164)," where the claimed "cache view" would be the referenced reconstructed database.

As per Applicant's argument that Hayashi does not teach "performing the read-only transaction using the read-only cache view and returning results of the read-only transaction, without blocking performance of transactions involving write operations using the write view of the given database," the Examiner agrees. The Examiner has instead cited Klein Fig. 10, col. 9, lines 5-17, "The retrieved data values from the rolled back transactions are provided (block 169)," and col. 7, line 66 – col. 8, line 8, "As would be readily recognized by one skilled in the art, a typical database engine could be concurrently processing multiple transaction operations at any given time. The described invention implicates read operations and does not directly effect the processing of other forms of database transactions."

As per Applicant's argument that Hayashi does not teach "creating a write view of a given database in the shared cache supporting read and write uses of the given database," the Examiner respectfully disagrees. Specifically, the Examiner cited Fig. 4 and col. 1, lines 30-41, "When a transaction accesses the database for some data through the database management system, the data may already be in a buffer shared by transactions due to another transaction that previously accessed the same data, or the data must be transferred from the database to the shared buffer," where the claimed "write view" is the referenced view of the "shared buffer."

Conclusion

Applicant's amendment necessitated the new ground(s) of rejection presented in this Office action. Accordingly, **THIS ACTION IS MADE FINAL**. See MPEP § 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the date of this final action.

Any inquiry concerning this communication or earlier communications from the Examiner should be directed to Aaron Sanders whose telephone number is 571-270-1016. The Examiner can normally be reached on M-F 9:00a-4:00p.

If attempts to reach the Examiner by telephone are unsuccessful, the Examiner's supervisor, Tim Vo can be reached on 571-272-3642. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

/Tim T. Vo/
Supervisory Patent Examiner, Art Unit
2168

/Aaron Sanders/
Examiner, Art Unit 2168
22 August 2008